

Table of Contents

Business Object Groups 2

[business object group](#), [business object](#), [concepts](#)

Business Object Groups

Business object groups in **AwareIM** offer you a simple and convenient way to handle similar yet different data.

Let us consider an application that registers client communication history, such as meetings, phone calls, letters, e-mails, etc. A summary of the history should be presented as a chronological list on the client form. The user should be able to see full details of any list item on a separate form.

Some details are common across all communications, like the contact time, summary or description. Other details are specific to a particular communication type, like the status of an outgoing e-mail can be Sent or Unsent, and the user should be able to prepare and send an e-mail, say by clicking a button. The status of an incoming e-mail can be Read or Unread. Phone conversations do not have any status at all. Letters should have a letter document attached to them, etc.

How would we represent communication details in our application? One way would be to combine the details of all communication types in a single business object called, say, `Communications`. The problem with the single business object is that depending on the communication type we would have to hide non-applicable details and operations from the data form or dynamically manipulate the values of the same attribute (like `Status` for incoming and outgoing e-mails). Alternatively, we could make several forms specific to each communication type. Then we would have to add some logic to all places where a communication form is displayed, like from the history list on the client form, to check the type and display an appropriate form. We would have to do the same check in all other places where the user can navigate to the communication form, like in a search result table.

Another way of going about it could be to create separate business objects for different communication types, like `Outgoing Emails`, `Incoming Emails`, `Outgoing Letters`, etc. Each of these objects would have its own data form. The trouble with separate objects is that it would be very difficult (if not impossible) to show records from separate objects in a single chronologically sorted list, say on a client form, or do a search across multiple objects.

With the solution that **AwareIM** offers you would create separate business objects for different communication types, like `OutgoingEmail`, `IncomingEmail`, `OutgoingLetter`, then create a business object group, say, called `Communication`, and include all the specific objects as its members. This is all it takes.

Each member of a group usually contains attributes common to all other members. In the example above the common attributes would be the date when the communication was sent or received and the state of the communication. Each member would also have some specific attributes not found in other group members, for example, telephone number for telephone communication; e-mail address for e-mail communication etc.

Business object groups are treated in the same way as regular business objects - they can be used in business rules, queries, document templates, and they also appear in the list of available attribute types. So, we would simply add a [multiple reference attribute](#) `ContactHistory` of type `Communication` to `Client` to display a list of communication records on the client form. When the user clicks on an item in the list, a form for the specific record will be automatically shown to the user.

Business object groups are useful in the following scenarios:

1. Business object groups can be used in queries to search the system for objects of different types based on the criteria that use common attributes of the group – see [Data Retrieval](#).
2. Business object groups can be referred to by reference attributes as shown in the above example – the ContactHistory attribute on the Client object refers to a group rather than to a particular object.
3. Business object groups can be used in business rules – again in this case rules may only use common attributes of the group. For example consider the following rules:

```
FIND Communication WHERE Communication.SentDate = CURRENT_DATE  
Communication.State = 'SENT'
```

Here the second rule would set the value of the state attribute of any communication object found by a query irrespective of its type.

tip

If business rules attached to different members of a group are exactly the same it is better to use group names in the rules rather than names of a particular group member. In this case changes made to a rule attached to one member will be automatically transferred to the equivalent rules attached to other members – see also [Adding/Editing Rules](#).

Configuration of business object groups is explained in the [Adding/Editing Business Object Groups](#) section.

From:
<http://www.awareim.com/dokuwiki/> - **Documentation**

Permanent link:
http://www.awareim.com/dokuwiki/docs/2000_concepts/0200_basics/0600_business_object_groups

Last update: **2025/06/12 02:09**

